

## 2 Skapa relationer

Microsoft Access är en relationsdatabas, vilket innebär att databasen består av ett antal tabeller som står i relation till varandra.

Du ska nu skapa en relation mellan de olika tabellerna i databasen. Relationer skapas antingen via kommandot **Relationer** i **Verktyg**-menyn eller genom att klicka på knappen »Relationer» i verktygsfältet.



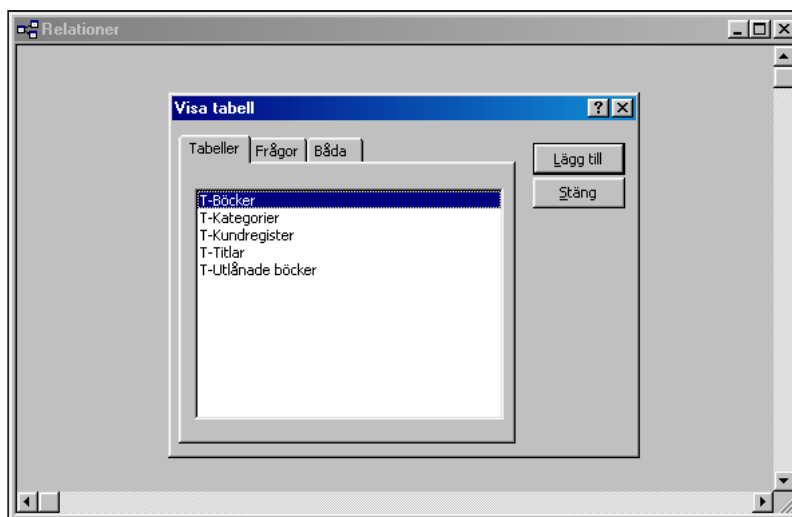
Relationer bör du skapa innan du anger data i dina tabeller. Då kan du inte skriva in felaktig information i tabellen utan att Access protesterar.

- 1 Se till att databasen **C:\Access2000-GK\Bibliotek** är öppen.
- 2 Välj **Relationer...** i **Verktyg**-menyn eller klicka på knappen »Relationer» i verktygsfältet.

Nu presenteras fönstret **Relationer**.

**FIGUR 4.22**

Här väljer du de tabeller som ska ha någon relation.



I detta fönster ska du lägga till de tabeller som du vill skapa relationer mellan.

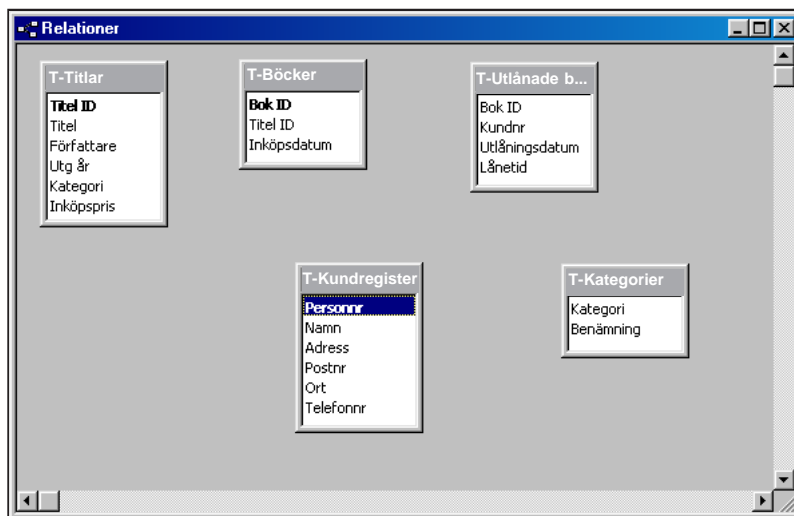
- 3 Om inte dialogrutan **Visa tabell** visas på skärmen väljer du **Visa tabell...** i **Relationer**-menyn.
- 4 I fönstret **Visa tabell** dubbelklickar du på alla tabellerna.
- 5 Stäng fönstret **Visa tabell** genom att klicka på knappen **Stäng**.

Tabellerna har nu lagts till i fönstret **Relationer**.

## 6 Placera dina tabeller enligt bilden nedan.

**FIGUR 4.23**

Placera ut tabellerna så här för att lättare skapa relationerna.

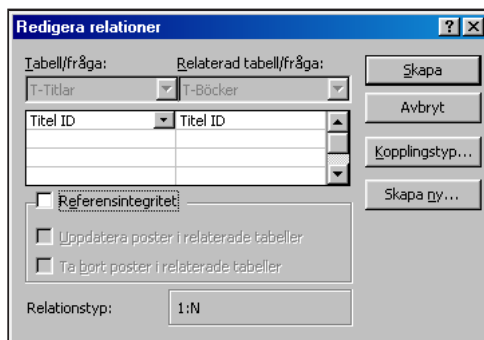


Som du ser har tre tabeller fet stil på ett fält, det betyder att fältet är definierat som Primärnyckel.

- 7 Klicka ned vänster musknapp i fältet **Titel ID** i tabellen **T-Titlar** och dra det till fältet **Titel ID** i tabellen **T-Böcker**. Släpp därefter musknappen och fönstret **Redigera Relationer** visas.

**FIGUR 4.24**

Fönstret **Redigera Relationer**.



Du ska nu skapa en *En-till-många* relation eftersom man kan ha många böcker med samma titel.

- 8 Markera rutan **Referensintegritet** för att aktivera det nedre fältet i dialogrutans. Klicka i valet **Ta bort poster i relaterade tabeller**. Detta gör vi för att få möjlighet att ta bort en bok från tabellen **T-Titlar** om boken t ex behövs tas bort från lagret.

## Referensintegritet

Referensintegritet är ett regelsystem som används för att säkerställa att relationer mellan poster i påverkade tabeller är giltiga och att relaterade data inte tas bort eller ändras av misstag. Du kan ange referensintegritet när samtliga av följande villkor är uppfyllda:

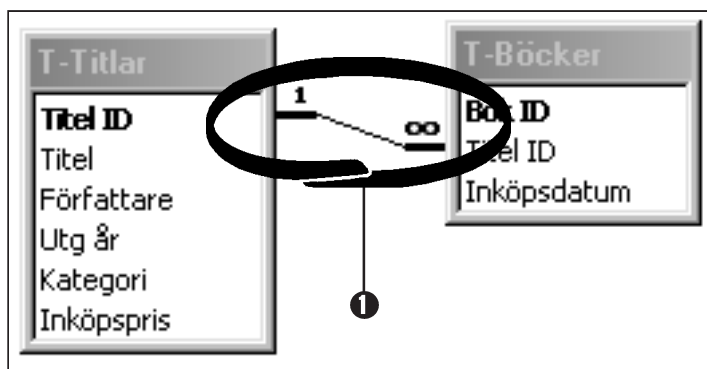
- Det fältet i din huvudtabell som ska ha en relation är en primärnyckel eller har ett unikt index.
- De relaterade fälten är av samma datatyp.
- Båda tabellerna finns i samma databas i Microsoft Access. Om tabellerna är länkade måste båda vara av samma databasformat.

Har du tabeller länkade från ett annan databasformat måste referenser skapas i den databasen som tabellerna skapats i. Det går inte att använda referensintegritet för länkade tabeller från olika databasformat.

9 Klicka nu på knappen **Skapa**.

**FIGUR 4.25**

*En-till-många relation, visas med en etta på ena tabellen och ett oändlighetstecken på andra tabellen.*



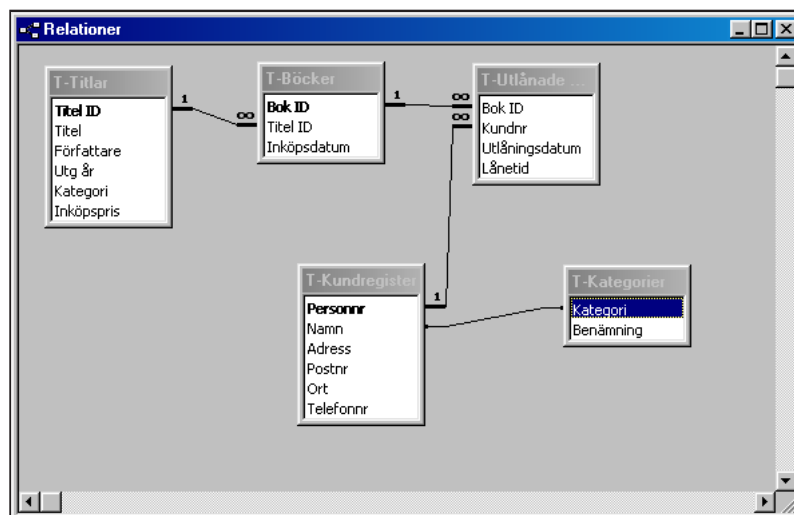
Nu har du skapat en *En-till-många* relation. Detta kan du även se på linjen som sammanbinder de två tabellerna med varandra. I den ena änden av linjen har du en **1:a** som symbol, i den andra änden ser du ett så kallat oändlighetstecken som symbol. Du kan också se att Referensintegritet är vald genom att linjen mellan de två tabellerna är smal och med en "klammer" till varje tabell.

- 10 Skapa nu de andra relationerna (se Figur 4.26), använd dig av referensintegritet men kryssa inte i **Ta bort poster i relaterade tabeller**.

Kopplingen mellan **T-Kundregister** och **T-Böcker** skapas endast i övnings syfte. Du skapar kopplingen på samma sätt som de andra med undantaget att du inte fyller i valet för **Referensintegritet**.

**FIGUR 4.26**

Här är alla tabellerna med deras olika referenser.



En förklaring till relationerna du precis skapat följer nedan.

- Relationen mellan tabellerna **T-Titlar** och **T-Böcker**.

Denna relation visar att ett **Titel ID** i tabellen **T-Titlar** kan förekomma många gånger i tabellen **T-Böcker**.

- Relationen mellan tabellerna **T-Böcker** och **T-Utlånade böcker**.

Denna relation visar att ett **Bok ID** i tabellen **T-Böcker** kan förekomma många gånger i tabellen **T-Utlånade böcker**. Eftersom du ska kunna låna ut en bok flera gånger.

- Relationen mellan tabellerna **T-Utlånade böcker** och **T-Kundregister**.

Denna relation visar att ett **Personnr** i tabellen **T-Kundregister** kan förekomma många gånger i tabellen **T-Utlånade böcker** (**Personnr=Kundnr**). En person ska kunna låna mer än en bok och det flera gånger.

- Kopplingen **T-Kundregister** och **T-Kategorier** har vi endast skapat i övnings syfte. Som du märker så är kopplingen inte logisk mellan de båda tabellerna.

- 11 Stäng relationsfönstret och välj att spara ändringarna.